CIS 787 – Analytical Data Mining

Fall 2015

Department of Computer Science, Syracuse University

Souradeep Sinha, MS in Computer Science – 536442648

α-Tweet

1. Aim

This project will aim at testing a given Tweet for probable number of Favorites and/or Retweets once before the user actually Tweets.

2. Data

The data shall be scraped from the user's Twitter account in real time. Hereby, we will call this user as the 0^{th} degree User or User. It shall consist of the 0_User's followers, tweets and timestamps for the last 28 days.

User's followers shall be called 1^{st} degree Users or Hop1Users. Our data will also consist of all the $Hop1User_i$'s Tweets, Retweets and Favorited Tweets for the last 28 days, i = {1, 2, … , number of User's followers}.

3. Plan of Action

Scrape Twitter data → Clean data using NLP APIs for only keywords → Apply Naïve Bayes principles for classifying tweet for a given follower → Evaluate model

4. Basic Idea

Calculate prior probability of User's tweet being favorited by $Hop1User_i$ as

$$P_{fav}(Hop1User_i) = \frac{Number\ of\ times\ Hop1User_i\ favorited\ User's\ tweet}{Number\ of\ tweets\ by\ User}$$

Find out the keywords in the tweet to be tested and store them in an array KeywordList containing $Keyword_j$ where j = {1, 2, … , length of KeywordList}. The keywords are important words that add the most amount of information to a sentence. Stop words such as "in", "is", "for", "the", "a", "an", "to", etc. will be filtered out and will not be a part of KeywordList.

Now, each such keyword is considered as an attribute of the tweet. We will calculate the probability of each keyword being favorited by a Hop1User as

$P_{fav}(Keyword_j \mid Hop1User_i) = \frac{Number\ of\ tweets\ favorited\ by\ Hop1User_i\ containing\ Keyword_j}{Number\ of\ tweets\ favorited\ by\ Hop1User_i}$

Therefore, $Keyword_j$ not favorited is

$\overline{P_{fav}}(Keyword_j \mid Hop1User_i) = 1 - P_{fav}(Keyword_j \mid Hop1User_i)$

Assuming the keywords being independent of one another (a huge assumption, given that words together can make up various contextual meanings), find out the posterior probability of a $Hop1User_i$ favoring tweet containing all the keywords in KeywordList $P_{fav}(Hop1User_i \mid KeywordList)$ (using the Naïve Bayesian method of calculating probabilities) as,

$$= \frac{P_{fav}(Keyword_1|Hop1User_i) \text{ x } \dots \text{ x } P_{fav}(Keyword_j|Hop1User_i) \text{ x } P_{fav}(Hop1User_i)}{some\ constant\ \rho}$$

Similarly, $Hop1User_i$ not favoring a tweet, $\overline{P_{fav}}(Hop1User_i \mid KeywordList)$

$$= \frac{(1- P_{fav}(Keyword_1|Hop1User_i)) \text{ x } \dots \text{ x } (1- P_{fav}(Keyword_j|Hop1User_i)) \text{ x } P_{fav}(Hop1User_i)}{some\ constant\ \rho}$$

If $P_{fav}(Hop1User_i \mid KeywordList) > \overline{P_{fav}}(Hop1User_i \mid KeywordList)$, a counter ProbableFavoriters will be incremented by 1.

We get the ProbableRetweeters counter by doing the same calculations, only this time accounting for retweets instead of favorites.

ProbableFavorites and ProbableRetweets are our final deliverables. They can be stored as a list of usernames to allow better insights.

Evaluation

Model can be evaluated by dividing the data as follows:
Training data – Data from 28th day to 7th day before present. (75% of data set.)
Testing data – Data from 7th day upto present day (25% of data set.)

$$\text{Error calculation} = \frac{|\text{Actual Favorited} - \text{ProbableFavorites}|}{\text{Actual Favorited}} + \frac{|\text{Actual Retweeted} - \text{ProbableRetweets}|}{\text{Actual Retweeted}}$$

Deliverable

Python source code
Input: User's username and the α-Tweet
Output: Probable number of favorites and retweets (maybe a list of such usernames)..